

## 超声波测距实验

### 【实验目的】:

- 1、掌握超声波测距的基本原理基本原理
- 2、学会 IO 的使用方法。

### 【实验环境】:

- 1、FS\_11C14 开发板
- 2、FS\_Colink V2.0
- 3、RealView MDK (Keil uVision4)

### 【实验步骤】:

- 1、测距文件夹下找到并打开 project.uvproj 文件;
- 2、编译此工程;
- 3、通过 FS\_Colink 下载编译好的工程到 FS\_11C14 开发板;
- 4、按 Reset 键复位, 观察 OLED 所显示的距离。

### 【实验现象】

1 接线说明:

红色----- (VCC) 5V

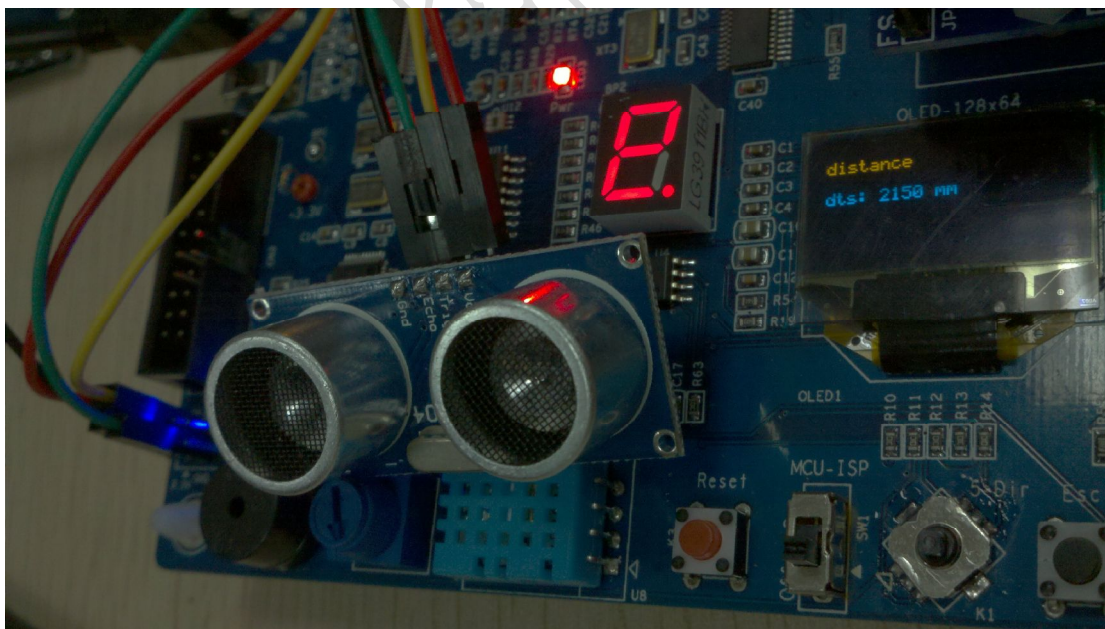
黑色----- (G) GND

绿色-----Echo (接 M0 实验板插针 J1 的 A)

黄色-----Trig (接 M0 实验板插针 J1 的 IO4)

用超声波探头测试距离, 在 2cm 到 45cm 之间。

图中测出超声波头与天花板的距离 2150mm,



### 【实验分析】

```
//led
```

```
GPIOSetDir(PORT2, 6, 1);
```

```
// Set PIO2_6 to output
```

```

GPIOSetValue(PORT2, 6, 0);    //

//配置采集为普通 IO，输入模式
LPC_IOCON->R_PIO1_2 &= ~0x07;
LPC_IOCON->R_PIO1_2 |= 0x01;    /* CLK OUT */

GPIOSetDir(PORT1, 2, 0);
GPIOIntDisable(PORT1, 2);

while(1)
{

    turn_flag = SysTick->VAL;
    GPIOSetValue(PORT2, 6, 1);
    if((turn_flag > 480) )//10us 高电平；使能信号
    { GPIOSetValue(PORT3, 0, 0);
      while(SysTick->VAL > (turn_flag-480));
      GPIOSetValue(PORT2, 6, 0);

    }else
    { GPIOSetValue(PORT3, 0, 0);
      while((SysTick->VAL > (48000+turn_flag-480))|(SysTick->VAL < turn_flag));
      GPIOSetValue(PORT2, 6, 0);
    }

    //获取传播时间
    GPIOSetInterrupt(PORT1, 2, 0, 0, 1);
    GPIOIntEnable(PORT1, 2);
    mo=0;
    cnt_last=p3_2_counter ;
    while(p3_2_counter == cnt_last);
    cnt_last = p3_2_counter;
    tc1 = p3_2_tc;
    cnt_mo = mo ;
    GPIOSetInterrupt(PORT1, 2, 0, 0, 0);
    while(p3_2_counter == cnt_last);
    cnt_last = p3_2_counter;
    cnt_mo2=mo;
    GPIOIntDisable(PORT1, 2);

    //时间统计
    nalve=cnt_mo2-cnt_mo;

```

```

        if(p3_2_tc < tc1)
        {
            if(nalve>0)
            {
                nalve=nalve-1;
                tc= 48000 - (p3_2_tc - tc1);
            }
            else
            { tc = tc1 - p3_2_tc;}
        }
        else
        {
            tc= 48000 - (p3_2_tc - tc1);
            nalve=nalve-1;
        }
        time_use = (nalve*1000+tc/48);
        distance=(uint32_t)( time_use*17/100);//计算距离

        printf(" distance=%d\r\n", distance);
        disbuf[0]= (uint8_t)(distance/1000);
        disbuf[1]=(uint8_t)(distance%1000/100);
        disbuf[2]=(uint8_t)(distance%100/10);
        disbuf[3]=(uint8_t)(distance%10);

        if(distance < 4500)
            snprintf(buf, 16, "dts: %d%d%d%d mm ", disbuf[0], disbuf[1],disbuf[2],disbuf[3]);
        OLED_DisStrLine(3, 0, (uint8_t *)buf);
        delay_ms(1000);
    }
}
else
    OLED_DisStrLine(3, 0, "Hitting ");
    delay_ms(250);
}

```